WHAT IS CLAIMED IS:

1. A method of aligning a boundary between bytes of a deserialized serial data signal that has been input into a programmable logic device, said programmable logic device having a serial data interface with a plurality of channels, and having a programmable logic core, said method comprising:

receiving a respective serial data signal on each respective channel of said serial data interface;

for each said channel, deserializing said serial data signal, including inserting an initial candidate byte boundary between selected bits of said deserialized data signal;

transmitting said deserialized data signal with said initial candidate byte boundary from each said channel of said serial data interface to said programmable logic core;

processing said deserialized signal with said initial candidate byte boundary on each said channel in said programmable logic core to validate each said initial candidate byte boundary; and

sending a respective byte alignment error signal from said programmable logic core to a respective channel of said serial data interface when said candidate boundary on said respective channel is determined to be invalid.

2. The method of claim 1 further comprising:

assigning based on said error signal an alternate candidate byte boundary between different selected bits of said deserialized data signal on said respective channel of said serial data interface;

retransmitting said deserialized data signal with said alternate candidate byte boundary from said respective channel of said serial data interface to said programmable logic core; and

reprocessing said retransmitted deserialized data signal with said alternate candidate byte boundary in said programmable logic core to validate said alternate candidate byte boundary.

3. The method of claim 2 wherein:

said error signal indicates a number of bits of discrepancy in said byte boundary; and

said assigning an alternate candidate byte boundary comprises moving said byte boundary by said number of bits of discrepancy.

4. The method of claim 2 wherein:

said assigning an alternate candidate byte boundary comprises moving said byte boundary by one bit; and

said sending, said assigning, said retransmitting and said reprocessing occur iteratively until said reprocessing determines that said alternate candidate byte boundary is correct.

5. A method of aligning a boundary between bytes of a deserialized serial data signal that has been input into a programmable logic device, said programmable logic device having a serial data interface with a plurality of channels, and having a programmable logic core, said method comprising:

receiving a respective serial data signal on each respective channel of said serial data interface;

for each said channel, deserializing said serial data signal, including inserting an initial candidate byte boundary between selected bits of said deserialized data signal;

transmitting said deserialized data signal with said initial candidate byte boundary from each said channel of said serial data interface to said programmable logic core; and

receiving a respective byte alignment error signal from said programmable logic core on a respective

channel of said serial data interface when said candidate boundary is determined in said programmable logic core to be invalid.

6. The method of claim 5 further comprising:

assigning based on said error signal an alternate candidate byte boundary between different selected bits of said respective deserialized data signal on said respective channel of said serial data interface;

retransmitting said deserialized data signal with said alternate candidate byte boundary from said respective channel of said serial data interface to said programmable logic core; and

again receiving a byte alignment error signal from said programmable logic core on said respective channel of said serial data interface when said candidate boundary is determined in said programmable logic core to be invalid.

7. The method of claim 6 wherein:

said error signal indicates a number of bits of discrepancy in said byte boundary; and

said assigning an alternate candidate byte boundary comprises moving said byte boundary by said number of bits of discrepancy.

8. The method of claim 6 wherein:

said assigning an alternate candidate byte boundary comprises moving said byte boundary by one bit; and

said receiving, said assigning and said retransmitting occur iteratively until said alternate candidate byte boundary is determined in said programmable logic core to be correct.

9. A programmable logic device comprising:

a programmable logic core; and

a serial data interface adapted to receive and deserialize a serial data signal including a plurality

of channels of serial data, said serial data interface comprising bit-slipping circuitry on each said channel adapted to a insert a byte boundary between bits of each respective deserialized data signal on each said channel, said bit-slipping circuitry being responsive to a bit-slipping control signal from said programmable logic core.

10. The programmable logic device of claim 9 wherein said bit-slipping circuitry comprises bit-handling circuitry and control circuitry that controls said bit-handling circuitry responsive to said bit-slipping control signal.

11. The programmable logic device of claim 10 wherein:

each byte includes a first number of bits; and

said bit-handling circuitry comprises:

at least one shift register for receiving said serial data, and having capacity to hold a second number of bits greater than said first number of bits, and further having a number of parallel outputs equal to said second number of bits, and

selection circuitry having a number of selection inputs equal to said number of parallel outputs of said at least one shift register, and having a number of selection outputs equal to said first number of bits, said selection inputs being connected to said parallel outputs of said at least one shift register; wherein:

said control circuitry controls which of said selection inputs is connected to said selection outputs.

12. The programmable logic device of claim 11 wherein said selection circuitry is a barrel shifter.

13. The programmable logic device of claim 11 wherein said second number of bits is twice said first number of bits.

14. The programmable logic device of claim 13 wherein said at least one shift register comprises two shift registers chained serially, each of said shift registers having capacity to hold said first number of bits.

15. A serial data interface for use with a programmable logic device having a programmable logic core, said serial data interface being adapted to receive and deserialize a serial data signal including a plurality of channels of serial data, said serial data interface comprising:

bit-slipping circuitry on each of said channels adapted to insert a byte boundary between bits of each respective deserialized serial data signal on each respective channel, said bit-slipping circuitry being responsive to a bit-slipping control signal from said programmable logic core.

16. The serial data interface of claim 15 wherein said bit-slipping circuitry comprises bit-handling circuitry and control circuitry that controls said bit-handling circuitry responsive to said bit-slipping control signal.

17. The serial data interface of claim 16 wherein:

each byte includes a first number of bits; and

said bit-handling circuitry comprises:

at least one shift register for receiving said serial data, and having capacity to hold a second number of bits greater than said first number of bits, and further having a number of parallel outputs equal to said second number of bits, and

selection circuitry having a number of selection inputs equal to said number of parallel outputs of said at least one shift register, and having a number

of selection outputs equal to said first number of bits, said selection inputs being connected to said parallel outputs of said at least one shift register; wherein:

said control circuitry controls which of said selection inputs is connected to said selection outputs.

18. The serial data interface of claim 17 wherein said selection circuitry is a barrel shifter.

19. The serial data interface of claim 17 wherein said second number of bits is twice said first number of bits.

20. The serial data interface of claim 19 wherein said at least one shift register comprises two shift registers chained serially, each of said shift registers having capacity to hold said first number of bits.

21. A programmable logic device comprising the serial data interface of claim 15.

22. A digital processing system comprising:

processing circuitry;

a memory coupled to said processing circuitry; and

a programmable logic device as defined in claim 21 coupled to the processing circuitry and the memory.

23. A printed circuit board on which is mounted a programmable logic device as defined in claim 21.

24. The printed circuit board defined in claim 23 further comprising:

memory circuitry mounted on the printed circuit board and coupled to the programmable logic device.

25. The printed circuit board defined in claim 24 further comprising:

processing circuitry mounted on the printed circuit board and coupled to the memory circuitry.

26. A digital processing system comprising:

processing circuitry;

a memory coupled to said processing circuitry; and

a programmable logic device as defined in claim 9 coupled to the processing circuitry and the memory.

27. A printed circuit board on which is mounted a programmable logic device as defined in claim 9.

28. The printed circuit board defined in claim 27 further comprising:

memory circuitry mounted on the printed circuit board and coupled to the programmable logic device.

29. The printed circuit board defined in claim 28 further comprising:

processing circuitry mounted on the printed circuit board and coupled to the memory circuitry.